IN THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A method for facilitating ~~server-initiated~~ communications between one or more application servers and one or more application clients using hyper-text transfer protocol (HTTP) ~~HTTP protocol~~ comprising ~~the steps of~~:

providing a communication server for one or more server-side applications in an HTTP based application server[[;]] , wherein the communication server receives notification message data from one or more of the server-side applications, wherein the notification message data received by the communication server is intended for one or more clients of the applications, and wherein the notification message data includes application message data, and wherein the communication server stores the application message data;

providing a communication client for one or more of the clients of applications in an HTTP based application client, wherein the communication client automatically generates polling requests to the ~~communications~~ communication server;

in response to receiving a polling request ~~the polling requests~~ from the communication client, the communication server determining whether any application message data has been stored that is intended for one or more of the clients of applications in the HTTP based application client, and if so, sending ~~any~~ the application message data to the communication client ~~that is intended for any clients of applications in the HTTP based application client~~; and

upon receiving the application message data, the communication client distributing the received application message data to the one or more clients of applications.

2. (Original) The method of claim 1, wherein the communication client parses the received application message data and distributes parsed data messages to the intended clients of the applications, which may cause the clients of applications to fetch information from corresponding servers of the application.

3. (Currently Amended) The method of claim 1, further comprising ~~the step of~~ providing a communication servlet coupled between the communication server and the communication client.

4. (Currently Amended) The method of claim 1, further comprising ~~the step of~~ providing a message buffer for storing the data received by the communication server from the applications.

5. (Original) The method of claim 4, wherein the message buffer is comprised of a hashtable.

6. (Original) The method of claim 5, wherein the hashtable is a two-tier hashtable.

7. (Original) The method of claim 1, wherein the clients are web-based clients.

8. (Original) The method of claim 1, wherein the message data includes instructions for fetching data from corresponding servers of the applications.

9. (Original) The method of claim 1, wherein the message data is used for direct consummation.

10. (Currently Amended) A client/server communication framework to facilitate ~~for facilitating server-initiated~~ communications to one or more clients using hyper-text transfer protocol (HTTP) ~~HTTP protocol~~ comprising:

a first server in an application server to send a first message to a second server in the application server, and also to provide ~~for providing~~ information to one or more clients using HTTP ~~protocol~~;

[[a]] the second server in the application server, coupled to the first server, to receive the ~~for receiving a~~ first message from the first server, to store the first message, and to send the first message to, ~~wherein the first message is intended to be sent to a first client in~~ an application client at a later time in response to receiving an HTTP polling request from the application client and determining that the first message was previously stored; and

~~a second client in~~ the application client to send the ~~for sending an~~ HTTP polling request to the second server, to receive ~~receiving~~ the first message from the second server, and to distribute ~~distributing~~ the first message to ~~the~~ a first client in the application client.

11. (Currently Amended) The client/server communication framework of claim 10, wherein the first server is a server for an application, the second server is a communication server, the

first client is a client for the application, and the application client further comprises ~~second client is~~ a communication client.

12. (Currently Amended) The client/server communication framework of claim 10, further comprising a memory location to store ~~for storing~~ messages received by the second server.

13. (Original) The client/server communication framework of claim 12, wherein the messages are stored in a hashtable.

14. (Original) The client/server communication framework of claim 10, wherein the first message includes information identifying the first client and the application.

15. (Currently Amended) The client/server communication framework of claim 10, further comprising;

a third server to provide ~~for providing~~ information to one or more clients using HTTP protocol, wherein the second server is coupled to the third server to receive ~~for receiving~~ a second message from the third server, wherein the second message is intended to be sent to a third client using HTTP protocol; and

wherein the second message is sent to the third client in response to the same or consecutive polling requests by the second client.

16. (Original) The client/server communication framework of claim 10, wherein the first server is an application in a web server, and wherein the one or more clients are web-based clients.

17. (Original) The client/server communication framework of claim 10, wherein the first message is used to instruct the first client to fetch information from the first server using HTTP protocol.

18. (Original) The client/server communication framework of claim 10, wherein the first message is consumed by the first client directly.

19. (Currently Amended) A method for facilitating ~~server-initiated~~ communications from one or more servers to one or more clients under hyper-text transfer protocol (HTTP), the method ~~HTTP protocol~~ comprising ~~the step of~~:

providing a first server to communicate ~~for communicating~~ with one or more clients;

providing a second server to receive ~~for receiving~~ a message from the first server and to store the message from the first server, wherein the message includes information intended to instruct ~~for~~ a first client to fetch data from the first server;

providing a second client in ~~the~~ a same application client ~~of~~ as the first client, wherein the second client is automatically to send ~~for sending~~ HTTP polling requests to the second server; and

upon receiving a polling request from the second client, the second server is to send ~~sending~~ the message from the second server to the second client; and

wherein the second client is to distribute ~~distributes~~ the message to the first client.

20. (Currently Amended) The method of claim 19, ~~further comprising the step of~~ wherein storing the message from the first server comprises storing the message into a buffer.

21. (Original) The method of claim 20, wherein the buffer is provided by a hashtable.

22. (Original) The method of claim 19, wherein the first server is an application under a web server, and wherein the one or more clients are web clients.

23. (Original) The method of claim 19, wherein the communications between the servers and clients uses HTTP protocol.

24. (Original) The method of claim 19, wherein the first client fetches data from the first server in response to the message.

25. (Original) The method of claim 19, wherein the first client consumes the message directly.

26. (Currently Amended) The method of claim 19, further comprising ~~the steps of~~:

providing a third server to communicate ~~for communicating~~ with one or more clients;

wherein the second server also is to receive ~~received~~ a second message from the third server, wherein the second message includes information intended for a third client;

upon receiving a polling request from the second client, sending the information intended for the third client to the second client; and

distributing the message from the second client to the third client.